8-5-2011

A Text Mining Application for Exploring the Voice of the Customer

Achim Botzenhardt
University of Mannheim, botzenhardt@eris.uni-mannheim.de

Andreas Witt
SAP AG, a.witt@sap.com

Alexander Maedche
University of Mannheim, maedche@eris.uni-mannheim.de

A Text Mining Application for Exploring the Voice of the Customer

Achim Botzenhardt
Chair of Information Systems IV (ERIS)
University of Mannheim,
D-68131 Mannheim, Germany
botzenhardt@eris.uni-mannheim.de

Andreas Witt
SAP AG, Dietmar-Hopp-Allee 16,
D-69190 Walldorf, Germany
a.witt@sap.com

Alexander Maedche
Chair of Information Systems IV (ERIS) and
Institute for Enterprise Systems (InES)
University of Mannheim,
D-68131 Mannheim, Germany
maedche@eris.uni-mannheim.de

ABSTRACT

Involving customers into the development process is a well-known best practice for creating successful products. The classical process of direct customer contact is time consuming and does not scale. In the past decade we have seen a tremendous growth in digitally available textual sources in the form of emails or web-based data such as communities and alike. These large collections of unstructured content are containing relevant customer input, however, humans can only deal with a limited amount of unstructured content in a given time. To address this problem, the concept of text mining has been established to help humans processing large amounts of unstructured content. In this paper we present first results of a design science research project. We have developed a text mining software artifact for extracting the voice of the customer from unstructured content with the goal of supporting the product development process. The artifact has been piloted and evaluated within a case study that we have carried out in cooperation with a large enterprise software company.

Keywords

Text Mining, Development Process, Customer Involvement, Design Science Research

INTRODUCTION

The importance of involving customers into product development has been researched for a long time (Brown and Eisenhardt, 1995). Customer involvement is not a trivial task and in general leads to a great amount and complexity of information to process. Besides direct customer contact, specifically textual sources are of great importance to hear the customers' voice. Textual sources typically come in form of emails and associated documents. Recently, we have seen more and more customers exposing their opinions and needs via social media platforms such as blogs and micro-blogs, forums and alike. Looking at large document collections reveals the strengths and weaknesses of human perception, compared to automated processing. On the one hand, humans can only deal with a limited amount of documents in a given time, while a computer can process large document collections extremely fast. On the other hand, a human reader can cope with the syntax, and semantic of a document, as long as it is written in a language familiar to the reader. This allows him/her not only to understand the meaning of the document, but even to read between the lines. In contrast, a computer can only handle documents which have been transformed to a processable, structured format. However, the possibility to analyze large document collections might reveal hidden patterns and therefore allow the user to gain additional knowledge. The goal of Text Mining is to combine the benefits of human perception and computational power. Text Mining can broadly be defined as knowledge-intensive process in which a user interacts with a document collection over time by using a suite of analysis tools. In a manner analogous to data mining, text mining seeks to extract useful information from data sources through the identification and exploration of interesting patterns. In the case of text mining, however, the data sources are document

collections, and interesting patterns are found not among normalized database records but in the unstructured textual data in the documents of these collections (Feldman and Sanger, 2006).

The potential of using text mining techniques specifically within the requirements engineering process has already been recognized by several authors (Falessi, Cantone and Canfora, 2010). Specifically in the elicitation and prioritization phase of the requirements engineering process huge potentials for text mining have been identified. Requirements Elicitation is concerned with learning and understanding the needs of users and project sponsors with the ultimate aim of communicating these needs to the system developers (Zowghi and Coulin, 2005). A successful requirements elicitation process is heavily reliant on the communication skills of requirement engineers and the commitment and cooperation of the system stakeholders (Zowghi and Coulin, 2005). It must therefore be the goal to remove communication barriers and enable agreement about the requirements by providing a knowledge basis as complete as possible. All kinds of available information sources, even if they consist of large amounts of unstructured textual data, should be considered. Similar to elicitation, the prioritization process relies on a comprehensive information basis to ensure solid decisions. Again, a source of information can be document collections like forums or e-mails. As an example, the ratio of volatility of a specific requirement might become obvious by analyzing a textual source over a certain time period. A highly volatile requirement might not be judged as important as a more stable one.

This paper is structured as following: We first present an overview on related work in the context of extracting information from unstructured content for supporting the software product development process. Building on the detailed related work analysis, we illustrate our research design including the research questions and the pursued research approach. Subsequently, we describe the established text mining framework, including the conceptual architecture, the development process and the implemented artifact. We have performed a case-study based evaluation and piloted the text mining framework in a software company. We describe the case study in detail as well as summarize and interpret the obtained qualitative evaluation results. Finally, we conclude the paper with a short summary and provide an overview on potential future work.

RELATED WORK

The extraction of information from unstructured content for supporting the software development process is becoming more and more important. The reason for this arises not only out of the growing amount of unstructured data (e.g. regarding the web), but also on the dependence of requirements engineering on documents written in natural language (Sawyer, Gacitua and Stone, 2008). Making advances by leveraging new tools or technology like artificial intelligence, linguistics, or information science has been recently designated as one of the major future research streams in requirements engineering (Cheng and Atlee, 2007).

A major part of research regarding the computer aided processing of unstructured content is mainly related to the support of early stages in requirements engineering which can be divided into several parts (Sawyer, Rayson and Cosh, 2005). On the one hand the creation of domain ontologies and on the other hand the creation of requirements engineering artifacts like UML diagrams with the help of NLP techniques.

Developing and understanding of a problem domain is a crucial part before the creation of the actual requirements. The usage of domain ontologies is a very common approach of getting such a specific knowledge. They support a knowledge engineer by providing him with the necessary information about the problem domain. This research stream has been coined ontology learning (Maedche and Staab, 2001). Several approaches on how to develop domain ontologies for supporting requirements elicitation out of unstructured content have been proposed, for example based on technical documents such as manuals and specifications (Kitamura, Hasegawa, Kaiya and Saeki, 2009). Especially unstructured content from the web offers a rich source for enhancing domain knowledge within the requirements elicitation process (Kaiya, Shimizu, Yasui, Kaijiri and Saeki, 2010). Kaiya et al. propose a method and a tool, which can mine general concepts from documents on the web and add such concepts to an existing domain ontology. The authors could show that such an enhanced ontology improves the quality of the requirements elicited, in terms of completeness and correctness, compared to a domain ontology which is not using additional web content. With respect to the creation of requirements engineering artifacts, natural-language processing techniques can be used to parse textual requirements descriptions and to generate corresponding semi-formal models, such as data flow diagrams and communication diagrams (Cheng and Atlee, 2007) or UML class diagram designs (Alkhader, Hudaib and Hammo, 2006).

Furthermore natural language processing and information retrieval techniques can be used to detect possible ambiguities and inconsistencies in textual or use case requirements (Cheng and Atlee, 2007). For example, existing information retrieval techniques can be used to statistically measure requirements similarity (Natt och Dag, Regnell, Carlshamre, Andersson and Karlsson, 2002; Cleland-Huang, Settimi, Romanova, Berenbach and Clark, 2007). In a nutshell, computer aided analysis is

offering great potential in identifying equivalent requirements and detecting relationships between requirements (Falessi, Cantone and Canfora, 2010).

As mentioned before, a number of researchers have investigated the application of automated information extraction from unstructured content. However, the potential of user generated web content with regard to requirements engineering is usually out of scope of such researchers. In contrast, other areas are already adopting this new information source. For example marketing, where consumer needs for new products are extracted from user generated content from the internet. This kind of information is often designated as voice of the customer. By using web mining, new product ideas can automatically be identified by extracting information from blogs (Thorleuchter, Van den Poel and Prinzie, 2010). Another approach is the mining of online customer reviews, which provides information on the opinions and experiences of thousands of customers (Zhang and Narayanan, 2010). All these information can also be used in the context of requirements engineering. One work which is already taking advantage of user generated web content is proposed by Kaiya et al. (Kaiya et al., 2010). Requirements analysis using blogs is another approach on taking benefit from unstructured content out of the web (Lange, 2008). The author investigates the application of several techniques like statistical text classification, task learning and social network analysis to the area of requirements elicitation. He found that the techniques mentioned above do provide great benefits to requirements engineering with respect to the elicitation of information from a large amount of stakeholders who were communicating in the internet.

However, there have also been some critical voices concerning the application of NLP in the area of requirements engineering (Ryan, 1993). The unanimous opinion is that natural language engineering techniques could not replace the human requirements analyst. But those techniques have reached a level of maturity, that offers great opportunities and support for product development (Kof, 2005; Sawyer, Rayson and Cosh, 2005; Sawyer, Gacitua and Stone, 2008). With our work we build on this assumption and try to establish a text mining framework that supports extracting the voice of the customer in the context of the product development process. Our major contribution is that our text mining framework has been applied and evaluated within a case study. By demonstrating real-world applicability we contribute to the existing knowledge base and demonstrate the usefulness and potential of applying text mining for supporting product development processes.

RESEARCH DESIGN

As mentioned above the goal of our work is to establish and evaluate a text mining framework supporting requirements engineering in a real-world environment. In this section we introduce the key research questions and the research approach we have been following in our work.

Research Question

Our overall research focuses on the possibility and boundary conditions for applying text mining for customer voice extraction as input for product development. We have identified the following three research questions as key drivers for the research work presented in this paper:

RQ1: Which processing steps need to be applied to extract the voice of the customer from unstructured content?

RQ2: What are the required capabilities and components of a software artifact that can be used in a real-world environment?

RQ3: What is the potential of text mining for extracting the voice of the customer from a qualitative and quantitative evaluation point of view?

RQ1 and RQ2 focus on the development of the framework from a process and implementation perspective. RQ3 focuses on the evaluation of text mining in a real-world environment.

Research Approach

We follow the Design Science Research (DSR) approach in our work. DSR is a research paradigm in which a designer answers questions relevant to human problems via the creation of innovative artifacts, thereby contributing new knowledge to the body of scientific evidence. The designed artifacts should be both useful and fundamental in understanding that problem (Hevner and Chatterjee, 2010). A framework which ensures the application of the three cycles is the design science research methodology defined by (Vaishnavi and Kuechler, 2008). Accordingly, design science research starts with the "Awareness of a Problem" phase. The subsequent phases - "Suggestion", "Development" and "Evaluation" - are normally performed iteratively during the course of the research project. By forcing back the design process to the "Awareness of Problem" phase, new constraints are defined, and the suggestion process is carried out building on these constraints. This is a

fundamental activity in the design process, because it updates the knowledge base by adjusting the original theories that informed the design process.

The work presented here in this paper represents the first iteration of the design science cycle. In this paper we focus on two major topics: First, we provide an overview on the implemented text mining framework including a process model and a software artifact. Second, we introduce a case study where we have piloted and evaluated our text mining framework.

A TEXT MINING FRAMEWORK FOR EXPLORING THE VOICE OF THE CUSTOMER

This section describes in detail the text mining framework. We first elaborate on the process model that has been applied. Second, we present the conceptual and technical architecture of the implemented artifact.

A Process Model for Text Mining of the Customer Voice

The Cross Industry Standard Process (CRISP) for Data Mining (Chapman, Clinton, Kerber, Khabaza, Reinartz, Shearer and Wirth, 1999) has been established as a quasi standard for data mining applications. CRISP segments the data mining process into several phases. These steps give structure to the development of the procedure model. CRISP distinguishes the phases of business understanding, data understanding, data preparation, modeling, evaluation and deployment.

Business Understanding. The initial phase focuses on understanding the project objectives and requirements from a business perspective, then converting this knowledge into a data mining problem definition and a preliminary plan designed to achieve the objectives (Chapman et. al, 1999). In our case the overall business objective of the project is to best possible fulfill customer needs. This is accomplished by correctly, timely and completely eliciting and prioritizing software artifact requirements. The business process which is responsible for this task is requirement engineering. To enable an effective RE process, the requirement engineers should be provided with any information relevant to their task. For this purpose, the overall data mining goal is to improve the gathering and prioritization of requirements by means of information extraction from internal and external textual sources. Depending on specific factors like the kind of product being build, the number of involved stakeholders and the type of software development model used in the organization, the amount and frequency of information needed for RE varies. The proposed text mining approaches only make sense if the organization possesses sources of textual data dealing with RE related topics. To assess the usability of these sources, one has to be clear about the desired kind of information to extract from them. Determining the needed data to meet the business objective is the phase of “Data Understanding".

Data Understanding. The data understanding phase starts with an initial data collection and proceeds with activities in order to get familiar with the data, to identify data quality problems, to discover first insights into the data or to detect interesting subsets to form hypotheses for hidden information (Chapman et. al, 1999). To perform the initial data collection, the data sources containing information relevant to the RE process, have to be determined. These are mainly sources containing customer interaction and the external “voice of the customer". Customer interaction represents information coming from unstructured interactions between representatives of the business and customers. The most relevant unstructured information source are emails. The “voice of the customer” represents no direct dialog with the customer. This includes mining of customer monologues available from surveys, discussion forums, or web logs. This kind of information is useful for discovering what customers think about the products.

Data Preparation and Modeling. The both phases of data preparation and modeling cover all activities to construct the initial dataset (data that will be fed into the modeling tool(s)) from the initial raw data. Data preparation tasks are likely to be performed multiple times and not in any prescribed order. Tasks include table, record and attribute selection as well as transformation and cleaning of data for modeling tools (Chapman et. al, 1999). The preparation of unstructured data is usually more challenging than the preparation of structured data. The figure below visualizes our approach that is based on three process steps and associated individual activities: 1) The extract-transformation-load step, 2) The information extraction step, 3) The analysis step.

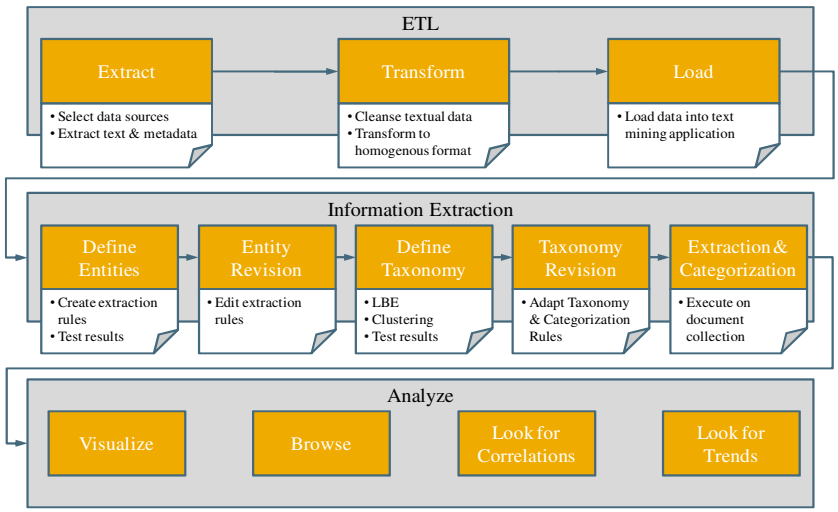

Figure 1. Data Preparation and Modeling activities

In the following we shortly explain the three steps:

- Extract – Transform – Load: Tasks include table, record and attribute selection as well as transformation and cleaning of data for modeling tools. The preprocessing methods have to be chosen according to storage location and original data format of the source text. The first step of data preparation is to extract the textual data (and if available the metadata) from the source. The extraction process should return a continuous character stream for each document which can be further processed. The approach most frequently found in text mining systems involves breaking the text into sentences and words, which is called tokenization. The main challenge in identifying sentence boundaries in the English language is distinguishing between a period that signals the end of a sentence and a period that is part of a previous token like Mr., Dr., and so on. After the documents have been tokenized, a feature vector for each document can be build by counting their tokens. The feature vector marks a specific position within the feature space of the document collection. The dimensions of the feature space are the words and phrases that occur in the information collection. The values for each dimension are the number of times each word or phrase occurs in each document. The documents are like locations on a map whose coordinates correspond to their word and phrase counts. After the documents have been transformed to a structured format which can be processed by the text mining application, the actual modeling of the data, like Information Extraction and Categorization can begin.
- Information Extraction: Information Extraction (IE) can be seen as a limited form of complete text comprehension. No attempt is made to understand the document at hand fully. Instead, one defines a priori the type of semantic information to be extracted from the document. IE represents documents as sets of entities and frames that are another way of formally describing the relationship between the entities. The first step in IE is to conduct a lexical analysis of the tokens. Each token is looked up in the dictionary to determine its possible part-of-speech and features. Afterwards names and other special forms, such as dates and currency amounts are being identified. Names are identified by a set of patterns (regular expressions) which are stated in terms of parts-of-speech, syntactic features, and orthographic features (Grishman, 1997).
- Analysis: In the analysis step various modeling techniques are selected and applied. One roughly can distinguish between supervised and unsupervised techniques. Supervised techniques operate based on given set of pre-classified data and create a model from this data. Examples for supervised techniques are decision trees, support vector machines, neutral networks, etc. Unsupervised techniques do not make any assumptions for the given data and explore structures and patterns. Examples for unsupervised techniques are clustering, association rules, etc.

*Evaluation***.** In the evaluation phase the generated models are checked against the business objectives. Evaluation from a quantitative perspective typically tries to compute quality measures such as classification accuracy in supervised learning. Beside the technical perspective, evaluation needs to be performed also on the business level, specifically focusing on the question if the achieved results create value and the most important business issues are covered. This is typically than following a qualitative approach, e.g. by carrying out expert interviews.

Quantitative evaluation of text mining is challenging because of the complexity of natural language. In the context of information retrieval, the key metrics of precision and recall have been established. They can be seen as extended versions of

accuracy, a simple metric that computes the fraction of instances for which the correct result is returned. Precision describes the exactness of the results, recall represents the completeness of the results. We think that a comprehensive text mining application should be evaluated from a quantitative perspective from more dimensions. According to (Rowley and Farrow, 2000) the characteristics of information are the following: objectivity, accessibility, relevance, currency, structure and organization and the system. All these properties influence the quality and usefulness of the extracted information. *Objectivity* can be further split into the concepts of Accuracy and Reliability. Accuracy means that data or information is correct. *Reliability* implies that the information is a true indicator of the variable that is intended to measure. The information characteristic *accessibility* is the availability of knowledge to potential users. *Relevance* of information is given if the information meets the user's requirements, and can contribute to the completion of the task in which the user is engaged. To contribute to its completion, the information must be delivered in an appropriate *granularity*, which means level of detail and completeness. The characteristic *currency* is related to the life cycle of information. Some information can be relatively stable, while other information is outdated very quickly. The *structure* of information should match the inherent structure of the corresponding discipline. The capability to categorize the extracted information in a similar way and represent relationships between these categories should be provided. A metric would be the quantity of featured extraction categories fitting the requirements specification in use. A basic example is the extraction of stakeholder information, which is certainly a category used in any requirements specification type. The remaining characteristic under investigation is the *system* through which information is communicated and stored. This information system includes people, hardware and software.

Based on this basic characterization we have derived a set of information quality metrics specifically targeting our application domain:

Information Characteristic		**Metric**
Objectivity	Accuracy	% of correct facts in result set
	Reliability	# of posts by contributor # of answers by contributor
Accessibility		# of documents / time period # of languages supported # of processable formats
Relevance	Completeness	% of relevant messages in result set
	Granularity	
Currency		publishing date vs. current date
Structure		# of categories fitting the requirements specification
Systems		Overall process time

Table 1. Information Characteristics and associated Metrics

Accuracy can be best judged by including expert knowledge and analyzing multiple sources. As an example, regarding an extracted complaint about a missing product feature, the product developer as an expert can judge whether the request is correct or not. Also a manual inquiry of other sources like product descriptions or user guides can be performed to check correctness. We define the percentage of correct facts in a given result set as a relevant metric for accuracy. Reliability implies that the information is a true indicator of the variable that is intended to measure. One example concerning the development process could be the measurement of customer satisfaction with a certain product feature. One way to analyze the reputation of someone posting a message in a forum is to measure her numbers of posts on the topic under investigation. To gain these metrics, it is necessary to analyze the metadata of the investigated source. Accessibility describes the availability of knowledge to potential users. Our text mining application is assisting the user to deal with the complex task of exploring large amounts of unstructured content. A reasonable indicator for the capability of the process to do so, is the amount and size of documents it can handle within a certain time period. Relevance is best judged by experts and a percentage for relevant messages within the result set can be returned. The metric for currency is the publishing date of the extracted information related to the current date. A possible metric for the structure would be the quantity of featured extraction categories fitting the requirements specification in use. Finally, the systems dimension may be measured in the overall process time.

Deployment. In the final phase, the results are deployed. This can happen in different forms. In the simplest case, the outcomes are documented in the form of a report. A more advanced way of deployment is embedded the process into a broader information systems, e.g. as done in the context of analytical customer relationship management (CRM), where data mining techniques are embedded into CRM systems to providing advanced decision support.

The Software Artifact

We have implemented a software artifact realizing the text mining process for extracting customer needs. We have built the software artifact based on two commercial software components: 1) SAP Business Objects Text Analysis XI 3.0[1] for entity and fact recognition, categorization, summarization and language identification. 2) Rapid Miner[2] including a set of data mining and exploration techniques. Figure 2 visualizes the key elements and data flows of the software artifact.

The software artifact converts source data into two pre-defined XML structures: one including the raw text and one including the metadata. The Text Analysis XI 3.0 component analyzes and categorizes text documents. In the context of text analysis it is possible to automatically identify and extract predefined entity types out of the text using the so-called ThingFinder. The ThingFinder comes with several predefined entity types relevant for voice of the customer extraction:

- Characteristic: Generic product characteristics and capabilities
- Problem: Major and minor problem statements
- Request: Existing, new and contact requests
- Sentiment: Strong/weak positive/negative and neutral sentiments

Besides the ThingFinder we have also used to so-called Categorizer. The Categorizer utilizes taxonomies created in the Categorizer Workbench. This component enables the development, management and testing of taxonomies and allows to define the rules by which documents are assigned to the taxonomies.

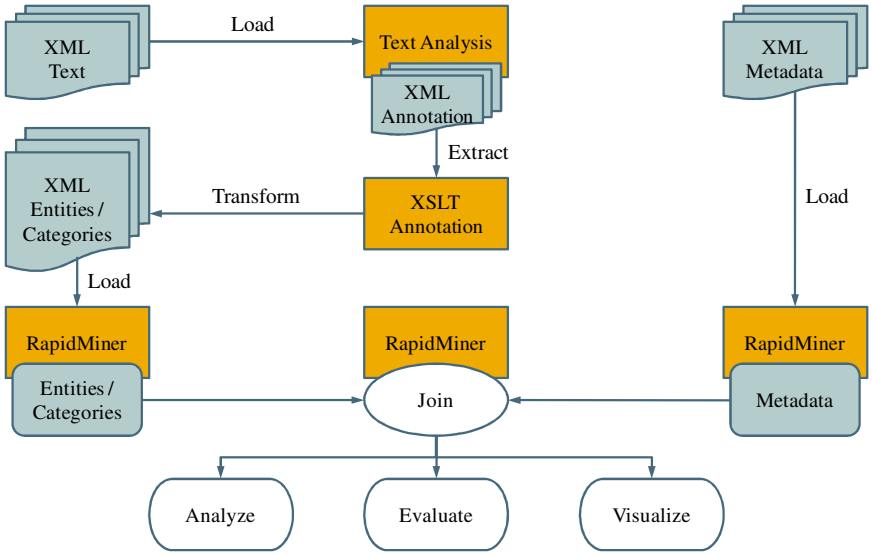

Figure 2. Software Artifact Components and Data Flow

The extracted entities from raw text as well as the metadata (e.g. the authors of the documents) are fed into Rapid Miner which enables detailed analysis, evaluation and visualization. Using RapidMiner we enable the computation and visualization of advanced metrics such as customer satisfaction based on the raw data delivered by the Text Analysis component. For example we have defined customer satisfaction with regards to a specific product category as following: For each document we count and weight the recognized entities representing the number of major problems, minor problems, strong/weak negative as well as positive sentiments. For one category we compute the mean customer satisfaction value of all documents relevant for this category. Figure 3 shows a screenshot from RapidMiner for the time-dependent evolution of customer

[1] http://www.sap.com/solutions/sapbusinessobjects/large/eim/textanalysis/index.epx
[2] http://rapid-i.com/content/view/181/196/

satisfaction of specific product group done in our case study. The positive and negative peaks in customer satisfaction correlate with new releases of the product category delivered to the market.

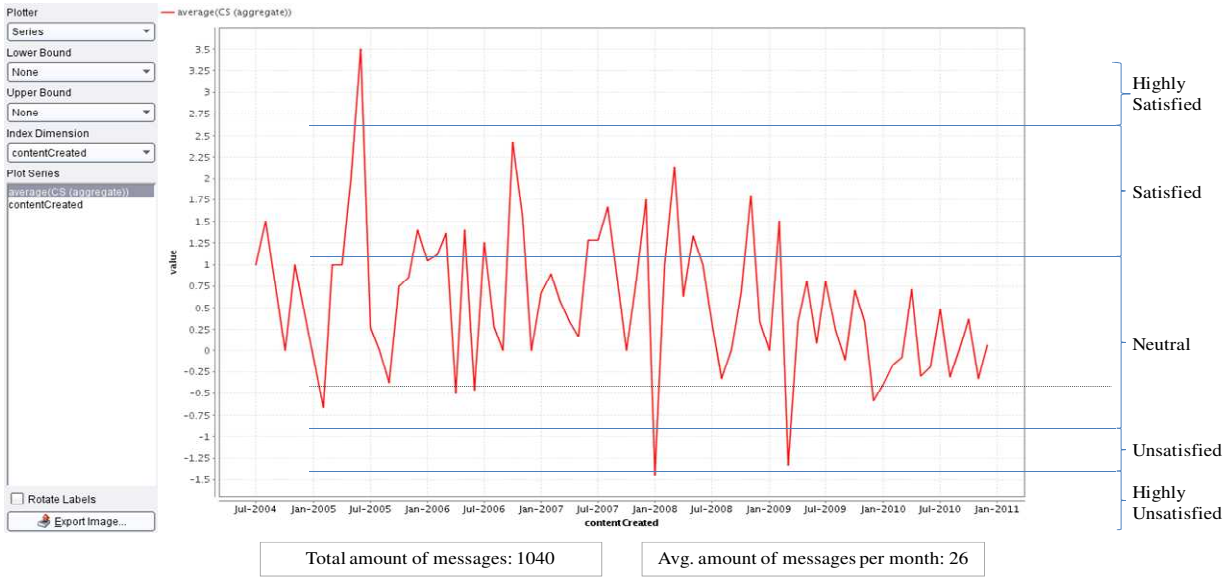

Figure 3. Customer Satisfaction of a product category over 6 years

CASE STUDY

The text mining application has been piloted and evaluated in a case study at the enterprise software company SAP AG. Major goal of this case study was to test feasibility and perform an evaluation from a quantitative and qualitative perspective. SAP AG is a software company that provides enterprise software applications and support to businesses of all sizes globally. Headquartered in Walldorf, Germany, with regional offices around the world, SAP is the largest enterprise software company in the world (as of 2009), it is also the largest software company in Europe and the fourth largest globally.

The case study has been carried out in cooperation within a product management unit responsible for a set of components of the SAP CRM product. We used two different data sources in our case study: First, we used an email collection containing CRM related information shared by consultants and developers. The test set which has been used is a PST-archive containing 2000 Outlook items. Second, we used the content contained in the forums contained in the SAP SCN Community (http://www.sdn.sap.com/irj/scn). It has over 2 million registered members, almost 29 million unique visitors in 2009 and about 6000 forum posts per day. SCN is public and therefore being used by various people like customers, partners, SAP experts and academics. The network is divided into multiple sub-communities, each one with a different focus. The ones most relevant for eliciting and prioritizing customer requirements are the SAP Developer Network (SDN) and the Business Process Expert (BPX). We specifically looked at entries with explicit CRM scope.

Results

We have performed a quantitative and qualitative evaluation. From a quantitative point of view we looked at the information characteristics introduced earlier in this paper. From a qualitative perspective, we have showed the text mining application to four product managers and interviewed them afterwards. In the following we present only selected quantitative and qualitative evaluation results of our case study.

The first information characteristic which has been evaluated is the accuracy of the extracted facts (entities). We specifically looked at the characteristic fact. A typical example for such an extracted fact would be the sentence: “BCM is offering a Chat channel." A test set of 1000 randomly selected documents (out of the email collection) has been used for the evaluation. Within the 1000 test documents, 591 sentences have been identified by the extraction rule as containing a characteristic fact. We have manually checked these suggestions: 156 of the 591 sentences included frequent standard phrases in emails which have been correctly identified as characteristics, but are irrelevant for product managers in their work. The remaining 435 extracted characteristic facts have been classified as being either correct or incorrect. The percentage of correctly identified characteristics is fairly low (54%). We did the same analysis for the facts requests, problems and sentiments. The

identification rates for request facts resulted in 61%, sentiments facts in 70% and for problems facts in 79%. The accuracy of the extracted fact types shows that the predefined extraction rule set is in principal beneficial in supporting the processes of extracting the customer voice from unstructured content. However, the accuracy needs further improvement specifically for characteristics and request facts.

Continuing with the judgment of the information quality we looked at the reliability of information. The reliability of the extracted information can best be judged by looking at the reputation of the contributor. It can be presumed that a contributor with high trustworthiness is likely to provide reliable information. Figure 4 visualizes the relationship between top contributors (> 10 entries in community) and customer satisfaction value. Thereby, each bar represents one author.

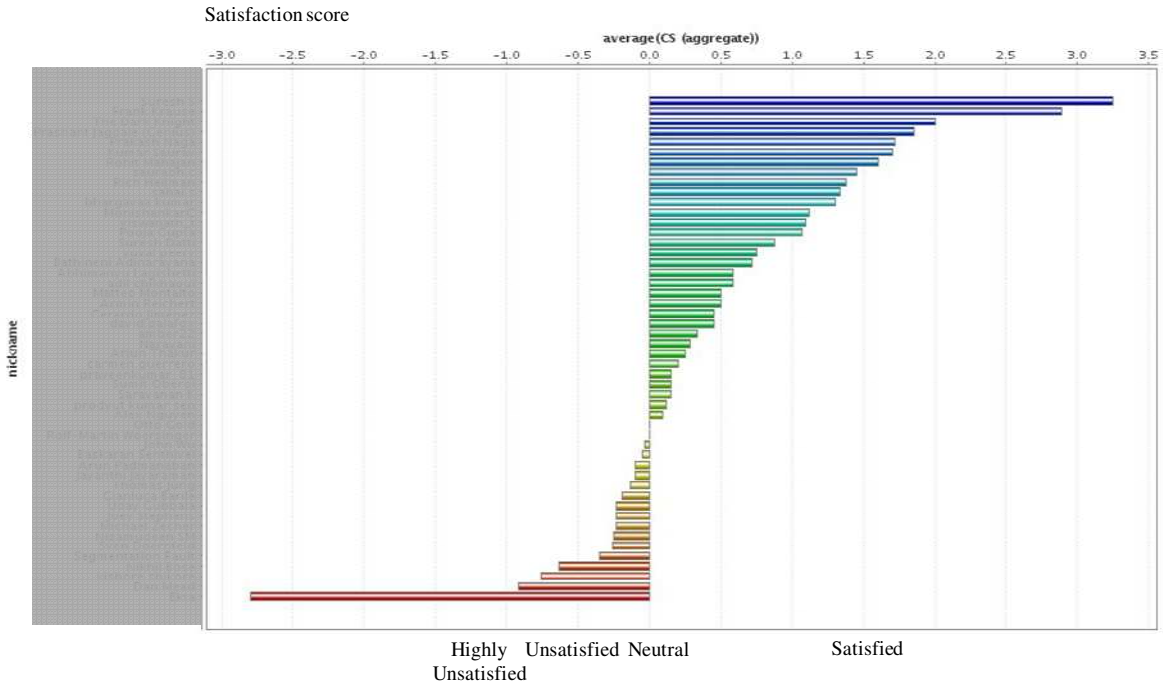

Figure 4. Top Contributors and Customer Satisfaction Index

The prototype offers ways of assessing the reputation using the processed metadata. In case of the data source being an email collection, the available metadata about the sender is his/her name and the amount and date of sent mails. Looking at the data source being the community, the prototype extracts more relevant metadata about the contributor's reputation. The most relevant is the amount of points she/he gained for successfully sharing knowledge in the forum. These points are given by the original poster to the contributor with the best answers provided. All these metadata are forwarded to the analysis tool RapidMiner and can be accessed to judge the reliability of the contributor.

In addition to the quantitative measurements, we have also carried out interviews with four product managers from the above mentioned CRM product management team to get feedback on our proposed text mining application. Among other questions, we specifically asked them for their first impression and satisfaction with the prototype as well as the usefulness and applicability of the prototype with regard to their job. The general feedback on the capabilities provided by the text mining application was positive. All of them confirmed the potential of the prototype to analyze unstructured content based on fact extraction in combination with metadata-based information such as the contributor as highlighted. Specifically the aggregation capabilities for large volumes of unstructured content were well received. One of the most interesting quotations by a product manager from our qualitative evaluation is the following:

> *"The good thing of such an application is that it helps preventing the personal bias which usually happens when exploring customer statements in unstructured content. The quantitative approach followed by text mining helps to scan large volumes of data and present the results in an aggregated form ..."*

CONCLUSION

The work presented in this paper represents the first iteration of a design science cycle. We have developed an integrated text mining application for extracting the voice of the customer from unstructured content with the goal of supporting the product development process. The artifact has been piloted and evaluated within a case study that we have carried out in cooperation with a large enterprise software company.

The contributions of our work are manifold. First, we enhanced the CRISP model with a text mining perspective. Second, our integrated text mining application combined text extraction capabilities with data mining capabilities and therefore is able to process unstructured content and correlate it with metadata. Third, we have defined a set of information characteristics and evaluated these characteristics using real-world data. The evaluation results have shown that extracting facts from unstructured data is a challenging task even for professional text mining tools. Nevertheless, our qualitative evaluation results demonstrate that text mining can help product managers in their daily work. We have discovered huge potential by using the key facts extracted from unstructured content to derive higher level key performance indicators such as the customer satisfaction index.

In our future work we plan to further research the fundamental design principles underlying text mining applications for supporting product development processes in general. Furthermore, we are also interested in the question how advanced analytical applications can be incorporated into information systems such as customer relationship management systems.

ACKNOWLEDGEMENTS

We would like to thank SAP AG for providing the possibility to pilot and evaluate the text mining application in a case study. Furthermore we thank Johannes Pfalzgraf for his contribution and support.

REFERENCES

1. Alkhader, Y., Hudaib, A. and Hammo, B. (2006) Experimenting With Extracting Software Requirements Using NLP Approach, *2006 International Conference on Information and Automation*, 349-354.
2. Brown, S. L. and Eisenhardt, K. M. (1995) Product Development: Past Research, Present Findings, and Future Directions, *The Academy of Management Review*, 20, 2, 343-378.
3. Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C. and Wirth, R. (1999) CRISP-DM 1.0 - Step-by-step data mining guide, www.crisp-dm.org.
4. Cheng, B. H. C. and Atlee, J. M. (2007) Research Directions in Requirements Engineering, *Future of Software Engineering (FOSE '07)*, 285-303.
5. Cleland-Huang, J., Berenbach, B., Clark, S., Settimi, R. and Romanova, E. (2007) Best Practices for Automated Traceability, *IEEE Computer*, *40*, 6, 27-35.
6. Falessi, D., Cantone, G. and Canfora, G. (2010) A comprehensive characterization of NLP techniques for identifying equivalent requirements, *Proceedings of the 2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement - ESEM '10.*
7. Feldman, R. and Sanger, J. (2006) The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data, Cambridge University Press.
8. Grishman, R. (1997) Information extraction: Techniques and challenges, in International Summer School on Information Extraction: A Multidisciplinary Approach to an Emerging Information Technology, pages 10-27, London, UK, Springer-Verlag.
9. Hevner, A. and Chatterjee, S. (2010) Design Research in Information Systems, Springer-Verlag Berlin.
10. Kaiya, H., Shimizu, Y., Yasui, H., Kaijiri, K. and Saeki, M. (2010) Enhancing Domain Knowledge for Requirements Elicitation with Web Mining, *2010 Asia Pacific Software Engineering Conference*, 3-12.
11. Kitamura, M., Hasegawa, R., Kaiya, H. and Saeki, M. (2009) A Supporting Tool for Requirements Elicitation Using a Domain Ontology, *Communications in Computer and Information Science*, *22*, 128-140.
12. Kof, L. (2005) Natural language processing: mature enough for requirements documents analysis?, in Andrés Montoyo, Rafael Muñoz and Elisabeth Métais (Eds.) Natural Language Processing and Information Systems, 91-102.
13. Lange, D. (2008) Text Classification and Machine Learning Support for Requirements Analysis Using Blogs, *Innovations for Requirement Analysis. From Stakeholders' Needs to Formal Designs*, 182-195.

14. Maedche, A. and Staab, S. (2001) Ontology Learning for the Semantic Web, *IEEE Intelligent Systems,* 16, 2, 72-79.
15. Natt och Dag, J., Regnell, B., Carlshamre, P., Andersson, M. and Karlsson, J. (2002) A Feasibility Study of Automated Natural Language Requirements Analysis in Market-Driven Development, *Requirements Engineering*, 7, 1, 20-33.
16. Rowley, J. and Farrow, J. (2000) Organizing Knowledge, Ashgate Publishing Limited, third edition.
17. Ryan, K. (1993) The role of natural language in requirements engineering, *Proceedings of IEEE International Symposium on Requirements Engineering,* January 4-6, San Diego, CA , USA, 240-242.
18. Sawyer, P., Gacitua, R. and Stone, A. (2008) Profiling and Tracing Stakeholder Needs, in Barbara Paech and Craig Martell (Eds.) Innovations for Requirement Analysis. From Stakeholders' Needs to Formal Designs, 196-213.
19. Sawyer, P., Rayson, P. and Cosh, K. (2005) Shallow knowledge as an aid to deep understanding in early phase requirements engineering, *IEEE Transactions on Software Engineering*, 31, 11, 969-981.
20. Thorleuchter, D., Van den Poel, D. and Prinzie, A. (2010) Extracting Consumers Needs for New Products - A Web Mining Approach, *2010 Third International Conference on Knowledge Discovery and Data Mining*, January 9, Phuket, Thailand, 440-443.
21. Vaishnavi, V. K. and Kuechler, W. (2008) Design Science Research Methods and Patterns, Auerbach Publications.
22. Zhang, K., Narayanan, R. and Choudhary, A. (2010) Voice of the customers: mining online customer reviews for product feature-based ranking, *Proceedings of the 3rd conference on Online social networks*, Berkeley, CA, USA, 11-11.
23. Zowghi, D. and Coulin, C. (2005) Requirements elicitation: A survey of techniques, approaches, and tools, in A. Aurum and C. Wohlin, Engineering and Managing Software Requirements, pages 19-46, Springer-Verlag Berlin.